Generate Collection Print

L5: Entry 1 of 4 File: TDBD May 1, 1984

TDB-ACC-NO: NN84056530

DISCLOSURE TITLE: Encryption Method for Providing Tagged Storage

PUBLICATION-DATA:
IBM Technical Disclosure Bulletin, May 1984, US

VOLUME NUMBER: 26
ISSUE NUMBER: 12
PAGE NUMBER: 6530 - 6536

PUBLICATION-DATE: May 1, 1984 (19840501)

CROSS REFERENCE: 0018-8689-26-12-6530

DISCLOSURE TEXT:

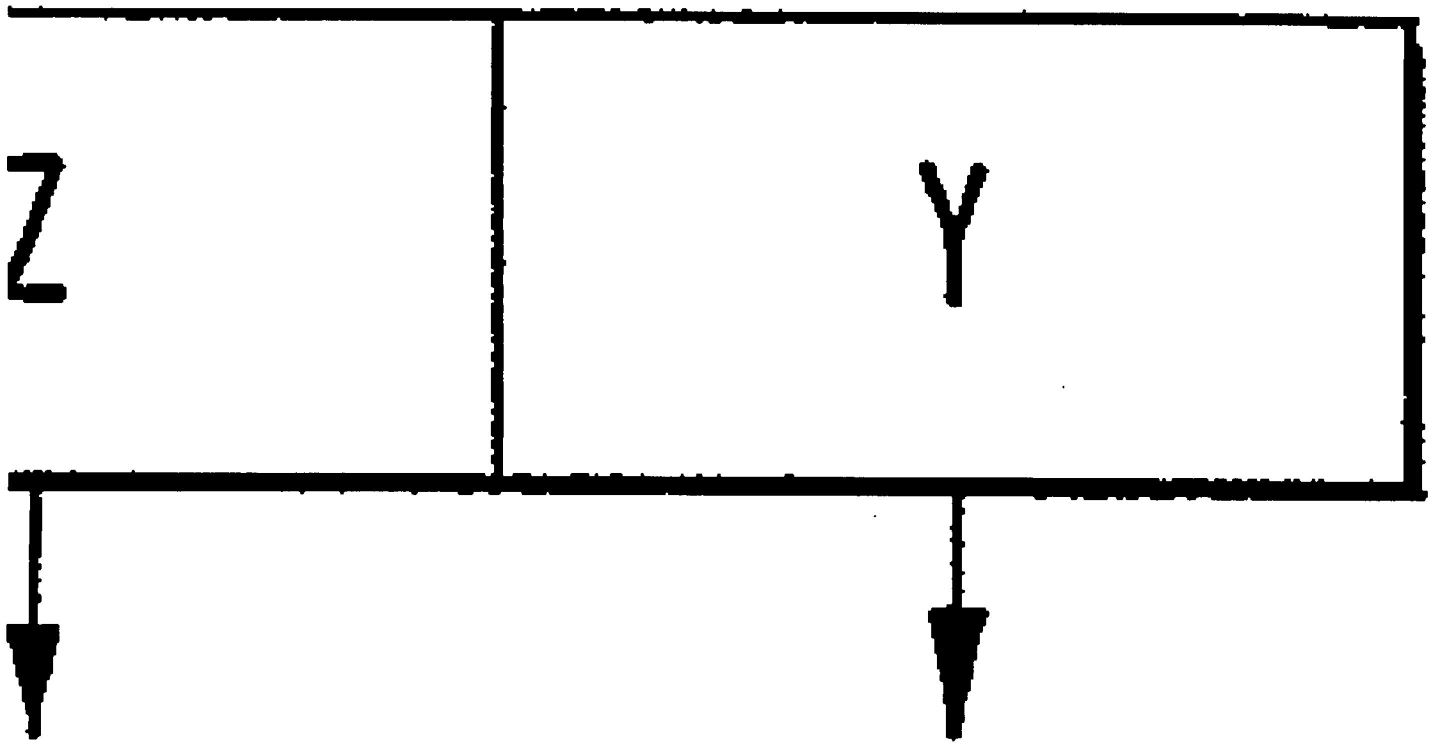
- The IBM System/38 has implemented "capability addressing". At the user interface, 16-byte pointers, called "system pointers", are constructed and presented to the Supervisor to perform high level functions. These system pointers represent address spaces, such as data base files, which the user cannot manipulate directly, but must use Supervisor-defined functions to manipulate. These system pointers may and often do include the authority to certain operations. Thus, the Supervisor uses the contents of pointers to validate function requests. This allows a flexible, high performance architecture. Similar 16-byte pointers control access to "Space Objects" which are linear address spaces. These address spaces contain 16-byte pointers and data together. The user has direct addressability to "Space Objects", allowing manipulation of multiple address spaces. Note that one option is to make a "Space Object" permanent. Unlike ordinary computational storage, pointers stored in "Space Objects" can be permanently located on Direct Access Storage Devices (DASDs). This form of addressing, where all permanent storage on the system is defined as addressable, segmented address spaces are called Single Level Store. The combination of Single Level Store and 16-byte capability pointers have been proposed for several architectures as well as being implemented on System/38. Other such architectures have come to similar conclusions about pointer size (16 bytes) and storing authority information in the pointer. This 16-byte pointer interface represents most of the protection mechanism of such architectures. That is, the ability to address storage represented by the 16-byte pointer is controlled by preventing the use of counterfeit pointers. The system requires a method of checking a pointer before loading it into a register capable of loading data storage. The remaining control is to prevent access to a limited region of each address space and to disallow a few instructions from being issued by the user. The usual solution to this problem is to implement a "tagged" main storage. Every 16-byte region of main storage has an associated bit which indicates whether or not the 16-byte area has a valid pointer. Ordinary arithmetic operations destroy this tag. Only special privileged Supervisor instructions may turn the tag bit on, creating a pointer. Thus, 16-byte pointers may only be created under Supervisor, and attempts to create counterfeit pointers are detected by checking the hardware tag bit. Because the storage of the tags in main storage and on the DASD is a major complication (especially in trying to use common hardware with non-tagged machines), it would be extremely desirable to "tag" the pointers in a different way. This method: - Solves the problem of tagging without using hardware tags. - Has almost the same performance characteristics. - Does not affect how

non-Supervisor code views the system. - With one exception, has the same security as the original hardware method. - Maintains position and time independence. Whether or not a pointer is valid does not depend on when it was constructed or its location. It is only dependent on the 16-byte pointer contents itself. The problem of tagging pointers without using hardware tags is twofold. First, the legitimate presentation of pointers must be cheaply validated. Second, the attempt to alter arbitrary bytes of storage to appear to be a valid pointer must be prevented. In particular, it must not be possible to take a pointer which contains particular kinds of authority, such as retrieving records, and upgrade the authority, such as to add insert authority. Another important aspect is that valid pointers must be permitted to be freely copied. The key notion of capability architecture is once a copy is given out with a given authority, such authority must be assumed given to the recipient for all time. Moreover, the recipient may copy it at will. This property is understood and expected by user level code. To limit the number of tag bits, 16 bytes has always been the size of such pointers, cryptography can be used. However, it is easy to compress the meaningful contents of a pointer into only 8 bytes. Two high-order bits would describe the pointer type (system, space, possibly others), 5 bytes less two bits would contain the Address Space Identifier. A 3-byte offset, either for the offset within the address space (for Space Objects) or for authority information (System Object Pointers), completing 8 bytes. This leaves an 8-byte area for a cryptogram to replace the hardware tag bit. A brief summary of relevant cryptography will lay the basis for how this can be done. In cryptographic authentication there is usually the following: - Some important data to be protected. - The need to protect not only the data, but to validate that the data came from a valid sender. The key to the solution is the existence of cryptographic algorithms with certain properties: - An encryption function, f, which is a function of a key x and arbitrary input y. Thus, z= f(x,y) forms a cryptogram z from key x using plaintext y. - For any proposed f there is a counterfeit encryption function F which performs z= F(y). That is, there is some function F which mimics the cipher without the key. Additionally, since both are reversible, there are inverse functions y= g(x,z) (the legitimate decryption function) and y= G(z) (the illicit decryption function) which reverse f and F. - The function G can be used by an opponent to illicitly obtain the plaintext y from cryptogram z. - Function F creates counterfeit cryptograms from plaintext y. - Cryptographic functions (such as the Data Encryption Standard (DES)) exist which have the property that "f" and "g" are reasonably efficient to compute and that "F" and "G" are computationally infeasible. For the Data Encryption Standard, this is believed to be true because the only known G is simply to try g(x,y) for all x until cryptogram z is observed. Since there are 2 to the 56th keys, and since 2 to the 55 average trials would normally be needed and 2 to the 49 or fewer trials would succeed less than one percent of the time, the system is asserted to be computationally infeasible to solve with foreseeable technology. As with all encryptions, someone may discover more efficient illicit F or G functions, but the system has withstood close scrutiny for a considerable period which allows as much confidence as can be had for a cryptosystem. - An additional restriction comes from the following. Every object created must have a pointer and must be tagged. If the encryption is system wide, then all users will have access to the function f(x,y'). Therefore, users can collect their own pairs z',y' in an attempt to discover someone else's z and y. Also possible is an attempt to deduce x. Once again, the Data Encryption Standard is believed to be invulnerable even to this level of attack, even with large numbers of z' and y' sets available. The remaining element of the solution is that Supervisor has an encapsulated portion of each address space which users cannot reference or display and that the 5-byte Address Space Identifier is unique over the life of the system. System/38 has several levels of authorization in the system pointer. Suppose, for simplification, there were only four levels. For example: None, Read, Write/Modify, Destroy. Suppose these levels were hierarchical such that destroy implied write, write implied read, and read implied none. Suppose a secure procedure existed for setting five system-wide secret keys (x1 thru x5). When Supervisor creates an object, it forms four cryptograms and places them in the encapsulated portion of the object: - z1 = f(x1,y), where y is a pointer to object y without authority. - z2 = f (x2,y), where y is a pointer with read authority. - z3 = f(x3,y), where y is a pointer with write/modify authority. - z4 = f(x4,y), where y is a pointer with destroy authority. - For Space Objects only, set z5 = f(x5,y), where y is a pointer of type

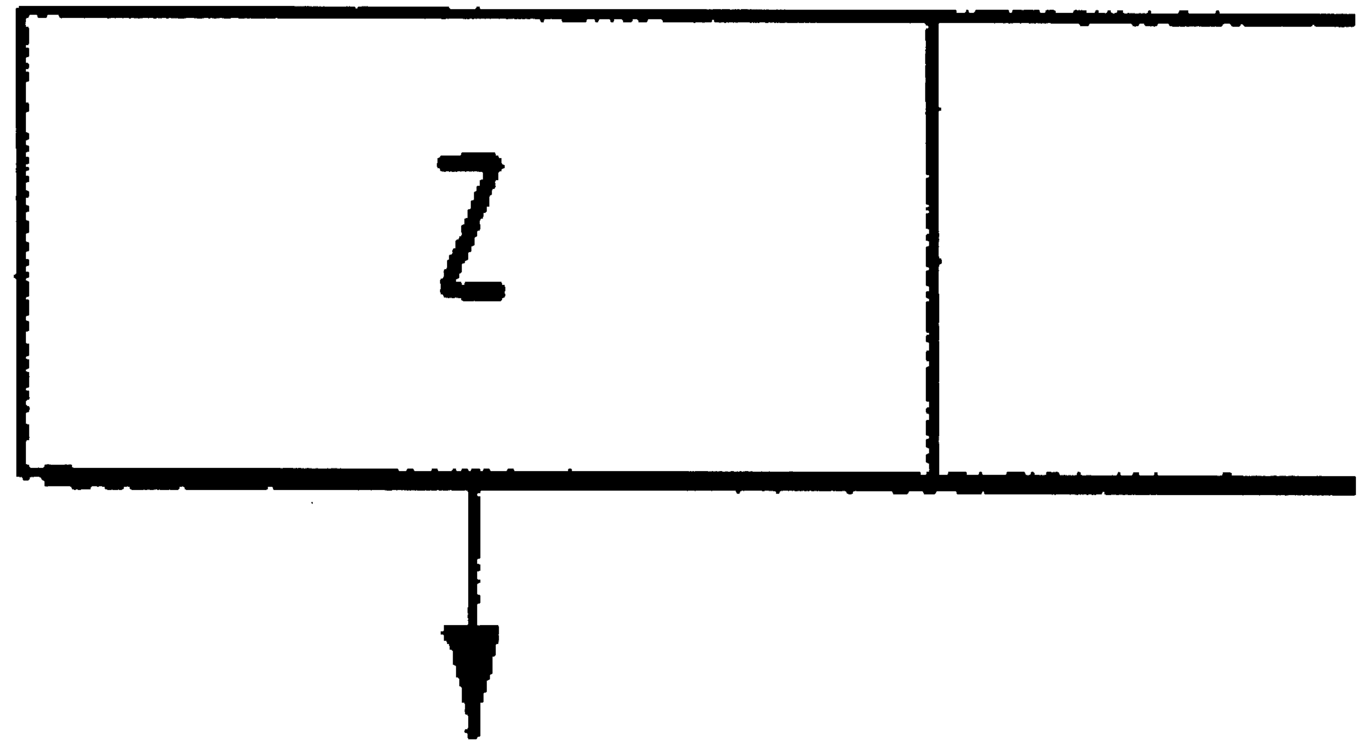
"space" (the others above had type "system"). When storing a System Pointer, Supervisor examines authority requested and copy z1, z2, z3, or z4 into the area reserved for the cryptogram/tag. Note that Supervisor may reject requests when the user asks Supervisor for more authority than the user is entitled to have. Space Pointers, when created, will always use cryptogram z5, regardless of offset. An encrypted pointer is shown in the figure below. ***** SEE ORIGINAL DOCUMENT ***** When loading a System Pointer, the system examines the authority in the pointer and selects a single cryptogram in the object header for comparison. The cryptogram selected is based on the highest level of hierarchical authority in the presented pointer. A match is given only if the cryptogram in the other portion of the pointer agrees with the selected cryptogram. The fact that other cryptograms may match, such as those with higher or lower hierarchical authority, is not checked. Space Pointers may be loaded after checking z5 and a bit indicating that the object is indeed a space object against the user-provided cryptogram. A hierarchical authority is not strictly required, but is a practical requirement. Even with only 4 authorities, 16 cryptograms would be required to have all of the authorities independent. This is already burdensome and rises to 32 if a fifth authority is added. Because more than one cryptogram is clearly required, a hierarchy of authorities is the natural choice. Assuming that the cryptograms are impossible to counterfeit, the following may be observed: - The Pointers may be freely copied. Since the cryptogram moves with the pointer, the pointer is just as valid in the copy as in the original. Likewise, a bogus pointer is not enhanced by moving it to some special place. - Pointers may not have their authority enhanced. In order to load a pointer with higher authority, a new cryptogram must also be presented. If one possesses a pointer, then one knows what its cryptogram is. However, if one possesses a pointer to an object with a low hierarchical authority, it is computationally infeasible to construct a pointer of greater authority. - Pointers may not be constructed. If one knows the low-order eight bytes of an object which one has no authority to, it is computationally infeasible to construct a pointer from scratch. For a system such as System/38: - All pointers must be validated before their first use. Note that not all uses of pointers must be validated; they might be loaded into an internal register and used multiple times. - No references must be made using pointers loaded by a program with superior authority. System 38 allows certain programs to adopt "User Profiles". This capability allows users to obtain pointers with authority which would otherwise be unavailable.

Such programs must not return to their caller with internally loaded pointers accessible. - No references must be allowed to the encapsulated part of the object which contains the cryptograms. - No references must be allowed to the secret keys x1 through x5. Key management is no different than the standard key management problem for a communications network. Since the user can only load pointers by presenting a 16-byte pointer, it is required that the user not be able to generate bogus pointers. One way a user might cause mischief is to alter the type bits to convert a system pointer to a space pointer. This would fool Supervisor into thinking the object was a space rather than an encapsulated object. However, since the plaintext is changed, the cryptogram must also change to successfully present the pointer. This is asserted as computationally infeasible when the DES is used. A user might turn authority bits on to improve hierarchical authority. However, the cryptogram must be altered. This is computationally infeasible. Revoking authority of a System Pointer can be done, but will not work unless the user has created a different pointer with less authority so the cryptogram may be copied. In such a case, the same interface could as easily be used. This case seems degenerate, but it is important to note for compatibility. If users clobber their own pointers, this should be detected and flagged. Revoking authority is one such alteration. Alteration of the cryptogram by the user does not solve the cipher to attempt to present fully authorized pointers because that is computationally infeasible. Instead, all possible cryptograms are presented directly. To defeat this attack, it is necessary only for Supervisor to assure that a particular exception, presentation of an invalid pointer, takes a finite time. Explicitly reading in a particular DASD sector, for example, will provide about 50 milliseconds or more, independent of Central Processing Unit (CPU) speed. Or a CPU loop of model dependent size may be executed to guarantee a certain speed. For 2 to the 63 average trials it would take 292 thousand years at one microsecond per trial. An important side note is

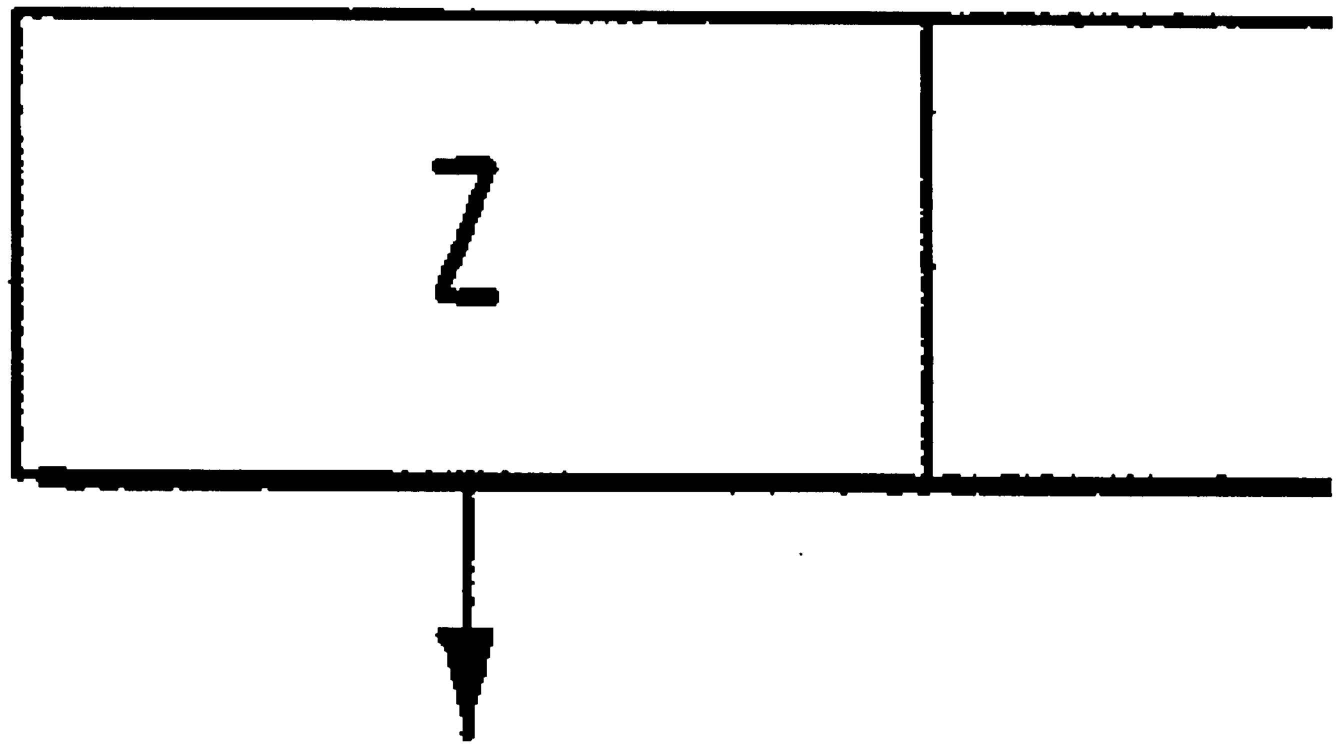
that it would be nearly impossible for an uninitialized pointer to be accidentally loaded, i.e., an attempt to load arbitrary bytes as a pointer due to a bug is for all practical purposes guaranteed to fail. Exhaustive DES attack is also possible but with an average of 2 to the 55th trials, 1100 years would be required at a microsecond per trial. Attempts to steal arbitrary files is also possible. Another approach is to essentially fix the cryptogram and try all possible plaintexts. The malicious user wants to wipe out "anybody's" file rather than a particular file, such as payroll. Statistically, this is the classic problem of removing balls from a well stirred urn without re placement. However, while the expression is easily written, the numbers (2 to the 64 factorial) are impossible to compute and Stirling's formula has too large an error for this calculation. However, it is possible to overestimate the malicious user's odds of success by approximating the large well stirred urn with a "smaller" well stirred urn with replacement. Such calculations show the costs are still on the order of 2 to the 63rd total trials. Attempts can be made to steal the image of a pointer. One feature of genuine hardware tagging which this solution cannot match is that of the "dump attack". If a malicious user obtains a storage dump with pointers displayed in hexadecimal form, he may copy such pointers and present them. With true hardware tagging, this attack is not possible. Accordingly, storage dumps have to be carefully structured and limited. However, such dumps are available only through service interfaces in System/38. As long as dumps are controlled, the system has no less of an exposure than it already has to various forms of invasion. For instance, if the malicious user is allowed service interfaces on today's System/38, pointers could be constructed, or more simply, authorities could be directly granted, obviating the need to illicitly create pointers. Given the enormous odds against arbitrary storage of appearing tagged when it is not, dump code of all kinds could present labeled information in place of detected tags, greatly limiting the problem. The "dump attack" is a weakness compared to true hardware tagging. System/38 has a translated interface, users cannot directly present instructions, but must have them generated from a program template. Thus, System/38 may support <u>encapsulated</u> space objects by controlling storage references in the generated code. Certain instructions in the current System/38 implementation assist this reference control. For machines which allow direct code generation, the mechanism works as follows: - Each storage reference has the high-order N bits fed to an N+1 input AND gate. The "extra" input would be the "privilege bit". This bit is a flip-flop which is turned on by any suitable mechanism, such as a System/370 SVC. If the address bits are zero and the inverse of the privilege bit is also zero, the reference is allowed. - Since single level store of some kind is implied, the load of a segment register would have to be privileged. One or two operations would be required. One would simply load the segment register with the Address Identifier. The second operation would use standard priority encoding circuits to implement "Load Cryptogram Address" from the contents of the eight bytes of pointer. Both operations could be combined into a single, privileged instruction.



?) GENERATED CRYPTOGRAM
OFFSET, AUTHORITY



SUPERVISOR(?) GENERATE
SEGMENT, OFFSET, AUT



SUPERVISOR(?) GENERATE
SEGMENT, OFFSET, AUT

WEST Search History

Hide Items | Restore | Clear | Cancel

DATE: Tuesday, April 13, 2004

Hide?	Set Name	Query	Hit Count
		DB=PGPB,USPT,USOC; PLUR=YES; OP=ADJ	
☐	L10	L7 and encapsula$	59
☐	L9	L7 and l4	7
		DB=EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=ADJ	
☐	L8	L7	1
		DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=ADJ	
☐	L7	L6 and widget	74
☐	L6	L5 and XML	976
☐	L5	validat$ near2 (data or input)	9021
		DB=PGPB,USPT,USOC; PLUR=YES; OP=ADJ	
☐	L4	717/100-123,126.ccls.	2167
☐	L3	20030070142	1
☐	L2	20020171690	1
☐	L1	20020171689	1

END OF SEARCH HISTORY

Search Results

Search Results for: **[XML<AND>(((data validation)))]**
Found **15** of **131,734 searched.**

Search within Results

GO > Advanced Search : > Search Help/Tips

Sort by: Title Publication Publication Date Score Binder

Results 1 - 15 of 15 short listing

1 Effective Web data extraction with standard XML technologies — 88%
Jussi Myllymaki
Proceedings of the tenth international conference on World Wide Web April 2001

2 Modelling stars using XML — 80%
Jaroslav Pokorny
Proceedings of the 4th ACM international workshop on Data warehousing and OLAP November 2001

We suppose collections of XML data described by Document Type Definitions (DTDs). This data has been generated by applications and plays a role of OLTP database(s). A star schema, a well-known technique used in data warehousing, can be applied. Then dimension information is supposed to be contained in XML data. We will use the notions of subDTD and view, and formulate referential integrity constraints in XML environment. We use simple pattern matching capabilities of current XML query languages ...

3 Babel: representing business rules in XML for application integration — 80%
Huaxin Zhang , Eleni Stroulia
Proceedings of the 23rd international conference on Software engineering July 2001

In this paper, we discuss Babel, a prototype tool for integrating multiple heterogeneous applications, by wrapping them and by specifying the logic of their interoperation in XML.

4 Software Performance Engineering of a Web service-based Clinical Decision Support infrastructure — 77%
ACM SIGSOFT Software Engineering Notes , Proceedings of the fourth international workshop on Software and performance January 2004
Volume 29 Issue 1

This paper has two contributions: a) it proposes a web services-based infrastructure to support Clinical Decision Support Systems (CDSSs) for processing multi-domain medical data from the obstetrical, perinatal and neonatal care domains, and b) applies Software Performance Engineering (SPE) to the proposed infrastructure. This extends a XML-based

framework for medical data interoperability and integration of CDSSs into the Neonatal Intensive Care Unit, developed previously by the authors. The fr ...

5 Document exchange model for augmenting added value of B2B collaboration 77%
Koichi Hayashi , Riichiro Mizoguchi
Proceedings of the 5th international conference on Electronic commerce September 2003
In this paper we present a B2B integration project, which aims to augment the added value of services instead of improving efficiency by automating processes. This paper introduces the VAlue Layered docUment Exchange (VALUE) model, which is a novel collaboration model adopted for this project. The key feature of the VALUE model is that partners exchange XML documents that are updated by adding elements for process information, such as the acceptance or rejection of a proposal, to received docume ...

6 Complex relationships and knowledge discovery support in the InfoQuilt system 77%
A. Sheth , S. Thacker , S. Patel
The VLDB Journal — The International Journal on Very Large Data Bases May 2003
Volume 12 Issue 1
Support for semantic content is becoming more common in Web-accessible information systems. We see this support emerging with the use of ontologies and machine-readable, annotated documents. The practice of domain modeling coupled with the extraction of domain-specific, contextually relevant metadata also supports the use of semantics. These advancements enable knowledge discovery approaches that define complex relationships between data that is autonomously collected and managed. The InfoQuilt ...

7 Session 3: Stylesheet transformations for interactive visualization: towards a Web3D chemistry curricula 77%
Nicholas F. Polys
Proceeding of the eighth international conference on 3D web technology March 2003
Recent Standards specifications offer important but underemployed techniques to maximize access-to and distribution-of information for real-time 3D visualization over the web. This paper describes and evaluates such techniques to transform structured data such as Chemical Markup Language (CML) to different forms and contexts for Web3D delivery using Extensible Stylesheet Transformations (XSLT), Extensible 3D (X3D), and VRML97. Standards design approaches offer a number of advantages: data durabi ...

8 At the forge: creating OpenACS packages 77%
Reuven M. Lerner
Linux Journal December 2002
Volume 2002 Issue 104

9 Industrial sessions: big data: The SDSS skyserver: public access to the sloan digital sky server data 77%
Alexander S. Szalay , Jim Gray , Ani R. Thakar , Peter Z. Kunszt , Tanu Malik , Jordan Raddick , Christopher Stoughton , Jan vandenBerg
Proceedings of the 2002 ACM SIGMOD international conference on Management of data June 2002
The SkyServer provides Internet access to the public Sloan Digital Sky Survey (SDSS) data for both astronomers and for science education. This paper describes the SkyServer goals and architecture. It also describes our experience operating the SkyServer on the Internet. The SDSS data is public and well-documented so it makes a good test platform for research

on database algorithms and performance.

10 B-trees: bearing fruits of all kinds 77%

Beng Chin Ooi , Kian-Lee Tan

Australian Computer Science Communications , Proceedings of the thirteenth Australasian conference on Database technologies - Volume 5 January 2002

Volume 24 Issue 2

Index structures are often used to support search operations in large databases. Many advanced database application domains such as spatial databases, multimedia databases, temporal databases, and object-oriented databases, call for index structures that are specially designed and tailored for the domains. Interestingly, in each of these domains, we find methods that are based on one distinct structure --- the B-tree. Invented some thirty years ago, the B-tree has been challenged repeatedly, but ...

11 The <bigwig> project 77%

Claus Brabrand , Anders Møller , Michael I. Schwartzbach

ACM Transactions on Internet Technology (TOIT) May 2002

Volume 2 Issue 2

We present the results of the <bigwig> project, which aims to design and implement a high-level domain-specific language for programming interactive Web services.

A fundamental aspect of the development of the World Wide Web during the last decade is the gradual change from static to dynamic generation of Web pages. Generating Web pages dynamically in dialog with the client has the advantage of providing up-to-date and tailor-made information. The development of systems ...

12 Principles for digital library development 77%

Alexa T. McCray , Marie E. Gallagher

Communications of the ACM May 2001

Volume 44 Issue 5

13 Implementing an e-commerce curriculum in a CIS program 77%

Rahul Tikekar , Daniel Wilson

The Journal of Computing in Small Colleges January 2001

Volume 16 Issue 2

In this paper we describe the development of an e-commerce curriculum in the CIS program at Southern Oregon University. The e-commerce curriculum is incorporated through a set of two courses. The first is a client-server programming course that introduces the structure of the web along with client and server side scripting techniques. The second is a corporate web development course, which builds on the earlier course by introducing elements of e-commerce technology such as security and encry ...

14 A reusable graphical user interface for manipulating object-oriented databases using Java and XML 77%

Suzanne W. Dietrich , Dan Suceava , Chakrapani Cherukuri , Susan D. Urban

ACM SIGCSE Bulletin , Proceedings of the thirty-second SIGCSE technical symposium on Computer Science Education February 2001

Volume 33 Issue 1

This paper describes the design and functionality of a graphical user interface (GUI) written in Java Swing that is used to support instructional activities associated with teaching object-

oriented database (OODB) concepts. The GUI supports the manipulation of objects in an OODB, assuming the implementation of a specified interface for interacting with an OODB. By using the interface, students can focus on object-oriented design and programming concepts associated with OODB concepts rather than ...

15 Managing the software design documents with XML 77%
Junichi Suzuki , Yoshikazu Yamamoto
Proceedings of the 16th annual international conference on Computer documentation
September 1998

Results 1 - 15 of 15 **short listing**

Search Results

Search Results for: **[widget<AND>(((data validation)))]**
Found **6** of **131,734 searched.**

Search within Results

GO > Advanced Search > Search Help/Tips

Sort by: **Title** **Publication** **Publication Date** **Score** **Binder**

Results 1 - 6 of 6 **short listing**

1 Programmimg pearls 77%
Jon Bentley
Communications of the ACM September 1985
Volume 28 Issue 9

2 Harvesting design for an application framework 77%
Joan Boone
Proceedings of the 1996 conference of the Centre for Advanced Studies on Collaborative research November 1996

Framework design begins with domain analysis. Either the problem domain is analyzed to create a new design, or the solution domain is analyzed to understand how the problem has already been solved. Solution domain analysis is advantageous for at least two reasons: solutions are likely to have addressed more concerns because they have been tested in a real-world environment and they provide a source for reusable code. On the other hand, such solutions may be too implementation-specific so as to i ...

3 Session 3: Stylesheet transformations for interactive visualization: towards a Web3D chemistry curricula 77%
Nicholas F. Polys
Proceeding of the eighth international conference on 3D web technology March 2003

Recent Standards specifications offer important but underemployed techniques to maximize access-to and distribution-of information for real-time 3D visualization over the web. This paper describes and evaluates such techniques to transform structured data such as Chemical Markup Language (CML) to different forms and contexts for Web3D delivery using Extensible Stylesheet Transformations (XSLT), Extensible 3D (X3D), and VRML97. Standards design approaches offer a number of advantages: data durabi ...

4 Linux for the International Space Station Program 77%
Guillermo Ortega
Linux Journal March 1999

An overview of two applications for spacecraft and why these applications are being run on

Linux

5 Expressing guidelines into an ergonomical styleguide for highly interactive applications — 77%

François Bodart , Jean M. Vanderdonckt
INTERACT '93 and CHI '93 conference companion on Human factors in computing systems April 1993

6 LISTEN: sounding uncertainty visualization — 77%

Suresh K. Lodha , Catherine M. Wilson , Robert E. Sheehan
Proceedings of the 7th conference on Visualization '96 October 1996

Results 1 - 6 of 6 **short listing**

Your search matched **24** of **1022101** documents.
A maximum of **500** results are displayed, **15** to a page, sorted by **Relevance** in **Descending** order.

Refine This Search:
You may refine your search by editing the current search expression or entering a new one in the text box.

validation<and>xml<and>data [Search]

☐ Check to search within this result set

Results Key:
JNL = Journal or Magazine **CNF** = Conference **STD** = Standard

---

1 **Logical definability and query languages over unranked trees**
Libkin, L.; Neven, F.;
Logic in Computer Science, 2003. Proceedings. 18th Annual IEEE Symposium on , 22-25 June 2003
Pages:178 - 187

[Abstract] [PDF Full-Text (384 KB)] **IEEE CNF**

---

2 **Embedding XML processing toolkit on general purpose programming language**
Kamina, T.; Tamai, T.;
Software Engineering Conference, 2002. Ninth Asia-Pacific , 4-6 Dec. 2002
Pages:159 - 166

[Abstract] [PDF Full-Text (259 KB)] **IEEE CNF**

---

3 **Realizing temporal XML repositories using temporal relational databases**
Amagasa, T.; Yoshikawat, M.; Uemura, S.;
Cooperative Database Systems for Advanced Applications, 2001. CODAS 2001. The Proceedings of the Third International Symposium on , 23-24 April 2001
Pages:60 - 64

[Abstract] [PDF Full-Text (372 KB)] **IEEE CNF**

---

4 **Using XML in relational database applications**
Malaika, S.;
Data Engineering, 1999. Proceedings., 15th International Conference on , 23-26 March 1999
Pages:167

[Abstract] [PDF Full-Text (60 KB)] **IEEE CNF**

---

5 **Constructing Web-based legacy index card archives-architectural design issues and initial data acquisition**
Downton, A.C.; Tams, A.C.; Wells, G.J.; Holmes, A.C.; Lucas, S.M.; Beccaloni, G.W.; Scoble, M.J.; Robinson, G.S.;
Document Analysis and Recognition, 2001. Proceedings. Sixth International Conference on , 10-13 Sept. 2001
Pages:854 - 858

[Abstract] [PDF Full-Text (408 KB)] **IEEE CNF**

---

6 **Towards self-validating knowledge-based archives**
Ludascher, B.; Marciano, R.; Moore, R.;
Research Issues in Data Engineering, 2001. Proceedings. Eleventh International Workshop on , 1-2 April 2001
Pages:9 - 16

[Abstract] [PDF Full-Text (648 KB)] **IEEE CNF**

---

7 **Metadata representation in XML for Internet-based electronic XML application from business to government**
Imamura, M.; Nagahama, R.; Suzuki, K.; Watabe, A.; Tsuji, H.;
Parallel and Distributed Systems: Workshops, Seventh International Conference on, 2000 , 4-7 July 2000
Pages:387 - 392

[Abstract] [PDF Full-Text (500 KB)] **IEEE CNF**

---

8 **The chamois component-based knowledge engineering framework**
Won Kim; Ki-Joon Chae; Dong-Sub Cho; Byoungju Choi; Anmo Jeong; Myung Kim; KiHo Lee; Meejeong Lee; Sang-Ho Lee; Seung-Soo Park; Hwan-Seung Yong;
Computer , Volume: 35 , Issue: 5 , May 2002
Pages:45, 47 - 54

[Abstract] [PDF Full-Text (1409 KB)] **IEEE JNL**

---

9 **XML schema language: taking XML to the next level**
Roy, J.; Ramanujan, A.;
IT Professional , Volume: 3 , Issue: 2 , March-April 2001
Pages:37 - 40

[Abstract] [PDF Full-Text (256 KB)] **IEEE JNL**

---

10 **Can XForm transform the Web? Transcending the Web as GUI.II**
Khare, R.;
Internet Computing, IEEE , Volume: 4 , Issue: 2 , March-April 2000
Pages:103 - 106

[Abstract] [PDF Full-Text (88 KB)] **IEEE JNL**

---

11 **Estimation of locally stationary covariance matrices from data**
Garcia, F.M.; Lourtie, I.M.G.;

Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03). 2003 IEEE International Conference on , Volume: 6 , 6-10 April 2003
Pages:VI - 737-40 vol.6

[Abstract] [PDF Full-Text (369 KB)] **IEEE CNF**

---

12 **Very fast transient over-voltages and transient enclosure voltages in gas insulated substations**
Amarnath, J.; Paramahamsa, D.R.K.; Narasimharao, K.; Singh, B.P.; Shrivastava, K.D.;
Electrical Insulation and Dielectric Phenomena, 2003. Annual Report. Conference on , 19-22 Oct. 2003
Pages:506 - 509

[Abstract] [PDF Full-Text (333 KB)] **IEEE CNF**

---

13 **A tool-supported method to extract data and schema from Web sites**
Estievenart, F.; Francois, A.; Henrard, J.; Hainaut, J.-L.;
Web Site Evolution, 2003. Theme: Architecture. Proceedings. Fifth IEEE International Workshop on , 22 Sept. 2003
Pages:3 - 11

[Abstract] [PDF Full-Text (364 KB)] **IEEE CNF**

---

14 **Metadata for the legal domain**
Palmirani, M.; Brighi, R.;
Database and Expert Systems Applications, 2003. Proceedings. 14th International Workshop on , 1-5 Sept. 2003
Pages:553 - 558

[Abstract] [PDF Full-Text (247 KB)] **IEEE CNF**

---

15 **A novel laboratory version management system for tracking complex biological experiments**
Shui, W.M.; Lam, N.; Wong, R.K.;
Bioinformatics and Bioengineering, 2003. Proceedings. Third IEEE Symposium on , 10-12 March 2003
Pages:133 - 140

[Abstract] [PDF Full-Text (350 KB)] **IEEE CNF**

Your search matched **24** of **1022101** documents.
A maximum of **500** results are displayed, **15** to a page, sorted by **Relevance** in **Descending** order.

Refine This Search:
You may refine your search by editing the current search expression or entering a new one in the text box.

validation<and>xml<and>data Search

☐ Check to search within this result set

Results Key:
JNL = Journal or Magazine **CNF** = Conference **STD** = Standard

---

1 **Logical definability and query languages over unranked trees**
Libkin, L.; Neven, F.;
Logic in Computer Science, 2003. Proceedings. 18th Annual IEEE Symposium on , 22-25 June 2003
Pages:178 - 187

[Abstract] [PDF Full-Text (384 KB)] **IEEE CNF**

---

2 **Embedding XML processing toolkit on general purpose programming language**
Kamina, T.; Tamai, T.;
Software Engineering Conference, 2002. Ninth Asia-Pacific , 4-6 Dec. 2002
Pages:159 - 166

[Abstract] [PDF Full-Text (259 KB)] **IEEE CNF**

---

3 **Realizing temporal XML repositories using temporal relational databases**
Amagasa, T.; Yoshikawat, M.; Uemura, S.;
Cooperative Database Systems for Advanced Applications, 2001. CODAS 2001. The Proceedings of the Third International Symposium on , 23-24 April 2001
Pages:60 - 64

[Abstract] [PDF Full-Text (372 KB)] **IEEE CNF**

---

4 **Using XML in relational database applications**
Malaika, S.;
Data Engineering, 1999. Proceedings., 15th International Conference on , 23-26 March 1999
Pages:167

[Abstract] [PDF Full-Text (60 KB)] **IEEE CNF**

---

5 **Constructing Web-based legacy index card archives-architectural design issues and initial data acquisition**
Downton, A.C.; Tams, A.C.; Wells, G.J.; Holmes, A.C.; Lucas, S.M.; Beccaloni, G.W.; Scoble, M.J.; Robinson, G.S.;
Document Analysis and Recognition, 2001. Proceedings. Sixth International Conference on , 10-13 Sept. 2001
Pages:854 - 858

[Abstract] [PDF Full-Text (408 KB)] **IEEE CNF**

---

6 **Towards self-validating knowledge-based archives**
Ludascher, B.; Marciano, R.; Moore, R.;
Research Issues in Data Engineering, 2001. Proceedings. Eleventh International Workshop on , 1-2 April 2001
Pages:9 - 16

[Abstract] [PDF Full-Text (648 KB)] **IEEE CNF**

---

7 **Metadata representation in XML for Internet-based electronic XML application from business to government**
Imamura, M.; Nagahama, R.; Suzuki, K.; Watabe, A.; Tsuji, H.;
Parallel and Distributed Systems: Workshops, Seventh International Conference on, 2000 , 4-7 July 2000
Pages:387 - 392

[Abstract] [PDF Full-Text (500 KB)] **IEEE CNF**

---

8 **The chamois component-based knowledge engineering framework**
Won Kim; Ki-Joon Chae; Dong-Sub Cho; Byoungju Choi; Anmo Jeong; Myung Kim; KiHo Lee; Meejeong Lee; Sang-Ho Lee; Seung-Soo Park; Hwan-Seung Yong;
Computer , Volume: 35 , Issue: 5 , May 2002
Pages:45, 47 - 54

[Abstract] [PDF Full-Text (1409 KB)] **IEEE JNL**

---

9 **XML schema language: taking XML to the next level**
Roy, J.; Ramanujan, A.;
IT Professional , Volume: 3 , Issue: 2 , March-April 2001
Pages:37 - 40

[Abstract] [PDF Full-Text (256 KB)] **IEEE JNL**

---

10 **Can XForm transform the Web? Transcending the Web as GUI.II**
Khare, R.;
Internet Computing, IEEE , Volume: 4 , Issue: 2 , March-April 2000
Pages:103 - 106

[Abstract] [PDF Full-Text (88 KB)] **IEEE JNL**

---

11 **Implementation of a novel virtual patient record architecture**
Berler, A.; Pavlopoulos, S.; Karkalis, G.; Sakka, E.; Konnis, G.; Koutsouris, D.;

[Engineering in Medicine and Biology, 2002. 24th Annual Conference and the Annual Fall Meeting of the Biomedical Engineering Society] EMBS/BMES Conference, 2002. Proceedings of the Second Joint , Volume: 3 , 23-26 Oct. 2002
Pages:1936 - 1937 vol.3

[Abstract] [PDF Full-Text (328 KB)] **IEEE CNF**

---

12 **Network middleware for utilization of sensors in room**
Noguchi, H.; Mori, T.; Sato, T.;
Intelligent Robots and Systems, 2003. (IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on , Volume: 2 , 27-31 Oct. 2003
Pages:1832 - 1838 vol.2

[Abstract] [PDF Full-Text (606 KB)] **IEEE CNF**

---

13 **A tool-supported method to extract data and schema from Web sites**
Estievenart, F.; Francois, A.; Henrard, J.; Hainaut, J.-L.;
Web Site Evolution, 2003. Theme: Architecture. Proceedings. Fifth IEEE International Workshop on , 22 Sept. 2003
Pages:3 - 11

[Abstract] [PDF Full-Text (364 KB)] **IEEE CNF**

---

14 **Metadata for the legal domain**
Palmirani, M.; Brighi, R.;
Database and Expert Systems Applications, 2003. Proceedings. 14th International Workshop on , 1-5 Sept. 2003
Pages:553 - 558

[Abstract] [PDF Full-Text (247 KB)] **IEEE CNF**

---

15 **A novel laboratory version management system for tracking complex biological experiments**
Shui, W.M.; Lam, N.; Wong, R.K.;
Bioinformatics and Bioengineering, 2003. Proceedings. Third IEEE Symposium on , 10-12 March 2003
Pages:133 - 140

[Abstract] [PDF Full-Text (350 KB)] **IEEE CNF**

---

1 2 Next

Searching for **data validation and widget and xml.**
Restrict to: Header Title Order by: Expected citations Hubs Usage Date Try: Amazon B&N Google (RI) Google (Web) CSB DBLP
No documents match Boolean query. Trying non-Boolean relevance query.
1000 documents found. **Only retrieving 250 documents (System busy - maximum reduced).** Retrieving documents... **Order: relevance to query.**

Artificial Intelligence for Decision Support: Needs.. - Miksch (1995) (Correct)
Artificial Intelligence, Knowledgebased System, **Data Validation**, **Data** Abstraction, and Intensive Care Intelligence, Knowledgebased System, **Data Validation**, **Data** Abstraction, and Intensive Care Medicine
ftp.ai.univie.ac.at/papers/oefai-tr-95-26.ps.Z

Wafe - A Tcl Interface to the X Toolkit or A Graphical.. - Neumann, Nusser (1996) (Correct)
:14 4.2.2 Resource Management over Resource **Data**base 14 4.2.3 Obtaining Type and cell. XbaeMatrix has callbacks for doing field **validation** and customizing 112 traversal. It allows the X Toolkit Release 4, 5 or 6 (Xt)the Athena **widget** set (Xaw) and (incomplete) to the OSF/Motif
nestroy.wi-inf.uni-essen.de/Lv/gui/wafedoc-a4.ps.gz

Integrating Temporal, Real-Time, and Active Databases - Ramamritham.. (1996) (Correct) (3 citations)
Integrating Temporal, RealTime, and Active **Data**bases Krithi Ramamritham, Raju Sivasankaran, John
www-ccs.cs.umass.edu/~sim/sigrec96.ps

The Data Reduction Expert Assistant - Miller (1992) (Correct)
1 The **Data** Reduction Expert Assistant Glenn E. Miller Space
www.stsci.edu/~miller/draco/draco-aldb.ps

Frames, Objects and Relations: Three Semantic.. - Norrie, Reimer.. (1994) (Correct)
for largescale knowledge base systems based on **data**base technologies and the three levels of seman
www.globis.ethz.ch/publications/docs/1994d-nrlrs-krdb.ps.gz

LOG: Building 3D User Interface Widgets by Demonstration - Matejic (1993) (Correct)
concepts the user intends without requiring precise **data** from the user. We say that LOG handles approximate LOG: Building 3D User Interface **Widgets** by Demonstration Larisa Matejic Department of
wilma.cs.brown.edu/pub/techreports/93/cs93-22.ps.Z

Scheduling Access To Temporal Data In Real-Time Databases - Xiong, Sivasankaran.. (1997) (Correct) (3 citations)
1 Scheduling Access To Temporal **Data** In Realtime **Data**bases Ming Xiong, Rajendran
www-ccs.cs.umass.edu/~sim/rtdb-chapter96.ps

Correction of a Memory Management Method for Lock-Free Data.. - Michael, Scott (1995) (Correct) (5 citations)
of a Memory Management Method for LockFree **Data** Structures \Lambda Maged M. Michael Michael L.
hypatia.dcs.qmw.ac.uk/data/edu/cs.rochester.edu/systems/95.tr599.Memory_management_for_lock-free_data_structures.ps.gz

Expanding Gaussian Kernels for Multivariate Conditional.. - Davis, Hwang (1998) (Correct) (1 citation)
in cases where the number of available training **data** points is relatively sparse within x space. We the kernel width, such as the leastsquare crossvalidation method [9]are also available with increasing
pierce.ee.washington.edu/~davisd/papers/tsp_kern.ps.gz

Swing Based Remote GUI Emulation - Tilley, Eklund (2000) (Correct)
1: Visualisation of a landbasin in 3D. A Spatial **Data**base Management System (SDBMS) under development by These are then rendered locally using Qt and **widget** callbacks are returned to the application via the
www.int.gu.edu.au/~ttilley/publications/Evolve2000.ps.gz

Designing GUIs by Sketch Drawing and Visual Programming - Eric Lecolinet (1998) (Correct)
instance an interactive graph editor) or to process **data** that changes at run time (for in stance a dynamic the system in order to produce the cor responding **widget** view (i.e. the actual visible GUI) and a graph of

www.infres.enst.fr/~elc/papers/avi98.ps

Implementing Fudgets with Standard Widget Sets - Reid, Singh (1998) (Correct) (4 citations)
calls to C language routines. Communicating **data** between Haskell and C programs is not trivial
event of an error (whether caused by failing the **validation** check or an unsuccessful query)an error shell
Implementing Fudgets with Standard **Widget** Sets Alastair Reid &Satnam Singh \Lambda
www2.cs.utah.edu/~reid/budgets.ps.gz

Statistical Learning, Localization, and Identification of.. - Hornegger, Niemann (1995) (Correct) (1 citation)
parameter estimation problems from projec ted **data**. The experiments show that this probabilistic
www5.informatik.uni-erlangen.de/TeX/Literatur/ps-dir/1995/Hornegger95:SLL.ps.gz

Mega-widgets in Tcl/Tk: Evaluation and Analysis - Jaeger (1995) (Correct)
Mega-**widgets** in Tcl/Tk: Evaluation and Analysis Shannon Jaege
www.cpsc.ucalgary.ca/grouplab/papers/1995/95-Megawidgets.TclWorkshop/megawidgets.pdf

Performance Comparison Of Video Transport Over ATM.. - Hossain, Kang, Horst (Correct)
display and control of the multimedia **data**. The control operation is done by sending rate
Y link X link ServerNet Links Main Memory 4K **Validation** &Translation Logic Address Figure 7:
berserk.vlsi.uiuc.edu/people/ashfaq/ieee.mm97.ps

Conjunctive Query Containment in Description Logics.. - Calvanese, De.. (1997) (Correct) (8 citations)
query con tainment is a central problem in several **data**base and knowledge base applications, in cluding
www.dis.uniroma1.it/pub/calvanese/calv-degi-lenz-DL-97.ps.gz

The Graph Widget - Implementation Manual - Beaudouin-Lafon (1992) (Correct)
:4 1.2 **Data** structures :
The Graph **Widget** implementation manual version 1.10 Michel
ftp.lri.fr/LRI/soft/ihm/graph/doc.ps.Z

An Overview of Document Mining Technology - Dixon (1997) (Correct)
as these hidden inside vast collections of text **data**, possibly giving companies that competitive edge
www.geocities.com/~mjdixon/mark/writings/dixm97_dm.ps

A Hypertext System for Integrating Heterogeneous, Autonomous.. - Noll, Scacchi (1994) (Correct) (2 citations)
Hypertext System (DHT)Based on a hy pertext **data** model and clientserver archi tecture, DHT
cwis.usc.edu/dept/ATRIUM/Papers/Integrating_Software_Repositories.ps

CSDC - The MoTiV Car Speech Data Collection - Langmann, Pfitzinger.. (1998) (Correct) (4 citations)
CSDC The MoTiV Car Speech **Data** Collection Detlev Langmann 1 ,Hartmut R.
from VERBMOBIL transliterations, if necessary. **Validation** The transcribers checked the **data**
www.phonetik.uni-muenchen.de/Publications/Pfitzinger_LREC98b.ps

First 20 documents Next 20

Try your query at: Amazon Barnes & Noble Google (RI) Google (Web) CSB DBLP

Hit List

Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Search Results - Record(s) 1 through 7 of 7 returned.

☐ 1. Document ID: US 20040015832 A1

Using default format because multiple data bases are involved.

L9: Entry 1 of 7 File: PGPB Jan 22, 2004

PGPUB-DOCUMENT-NUMBER: 20040015832
PGPUB-FILING-TYPE: new
DOCUMENT-IDENTIFIER: US 20040015832 A1

TITLE: Method and apparatus for generating source code

PUBLICATION-DATE: January 22, 2004

INVENTOR-INFORMATION:

NAME	CITY	STATE	COUNTRY	RULE-47
Stapp, Michael	Westford	MA	US	
Morgan, Robert	Concord	MA	US	

US-CL-CURRENT: 717/106; 717/149

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | Claims | KWIC | Draw Desc | In

☐ 2. Document ID: US 20030145305 A1

L9: Entry 2 of 7 File: PGPB Jul 31, 2003

PGPUB-DOCUMENT-NUMBER: 20030145305
PGPUB-FILING-TYPE: new
DOCUMENT-IDENTIFIER: US 20030145305 A1

TITLE: Method for developing and managing large-scale web user interfaces (WUI) and computing system for said WUI

PUBLICATION-DATE: July 31, 2003

INVENTOR-INFORMATION:

NAME	CITY	STATE	COUNTRY	RULE-47
Ruggier, Mario	Segny		FR	

US-CL-CURRENT: 717/100

ABSTRACT:

A method for developing and managing large-scale web user interfaces (WUI), comprising: designing and implementing the WUI using the Model View Controller (MVC) paradigm, and adding an additional layer (5) to the MVC paradigm, said additional layer (5) comprising a UI description (1), said UI description comprising all UI elements that are used by at least one of the MVC implementation components, such as views and the sub-elements used to define them, descriptions of content data, and the behavior of the WUI.

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | Claims | KWIC | Draw Desc | Ir

---

☐ 3. Document ID: US 20020080200 A1

L9: Entry 3 of 7 File: PGPB Jun 27, 2002

PGPUB-DOCUMENT-NUMBER: 20020080200
PGPUB-FILING-TYPE: new
DOCUMENT-IDENTIFIER: US 20020080200 A1

TITLE: Method and apparatus for implementing a web application

PUBLICATION-DATE: June 27, 2002

INVENTOR-INFORMATION:

NAME	CITY	STATE	COUNTRY	RULE-47
Wong, Garland	Rancho Santa Fe	CA	US	
Chue, Carlos	San Diego	CA	US	
Tang, Anthony	San Diego	CA	US	
Ghanbari, Reza	San Diego	CA	US	
Calire, Dyami	Encinitas	CA	US	
VanLydegraf, Eric	Solana Beach	CA	US	
Sanchez, Meliza P.	Carlsbad	CA	US	
Barnes, Trent	Vista	CA	US	

US-CL-CURRENT: 717/100

ABSTRACT:

In one embodiment, a method is described. The method of designing a web application includes designing a set of components, each component having a set of instances. The method also includes designing an application having references to the set of components. The method further includes designing an interface having references to the application, and building the application based on the interface and the designing of the application.

In an alternate embodiment, a method is also described. The method of providing a web-based application includes receiving a request for a web-based application. The method further includes accessing the web-based application. The method also includes accessing a set of objects related to the web-based application within a repository, and executing the web-based application including the set of objects in a manner including interaction with a requestor originating the request for the web-based application.

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | Claims | KWIC | Draw Desc | Ir

☐ 4. Document ID: US 6601234 B1

L9: Entry 4 of 7 File: USPT Jul 29, 2003

US-PAT-NO: 6601234
DOCUMENT-IDENTIFIER: US 6601234 B1

TITLE: Attribute dictionary in a business logic services environment

DATE-ISSUED: July 29, 2003

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Bowman-Amuah; Michel K.	Colorado Springs	CO		

US-CL-CURRENT: 717/108; 705/7, 717/107, 717/116

ABSTRACT:

A system and method are provided for controlling access to data of a business object via an attribute dictionary. The attribute dictionary, which stores attribute names and values, is dispatched over a network. A helper facade is provided for interfacing a business object and the attribute dictionary. Next, it is verified that a current user is authorized to either set or get one of the attribute values upon a request which includes the attribute name that corresponds to the attribute value. The helper facade is called to set, get, or update one of the attribute values based on the corresponding attribute name, wherein the helper facade shields the attribute dictionary from the application code of the business object. The attribute value in the attribute dictionary is obtained or updated if the verification is successful, and a dirty flag is set in the attribute dictionary and an indicator is broadcast upon the attribute value being updated.

15 Claims, 195 Drawing figures
Exemplary Claim Number: 1
Number of Drawing Sheets: 123

☐ 5. Document ID: US 6550057 B1

L9: Entry 5 of 7 File: USPT Apr 15, 2003

US-PAT-NO: 6550057
DOCUMENT-IDENTIFIER: US 6550057 B1

TITLE: Piecemeal retrieval in an information services patterns environment

DATE-ISSUED: April 15, 2003

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY

Bowman-Amuah; Michel K.	Colorado Springs	CO		

US-CL-CURRENT: 717/126; 700/80, 707/5, 717/101, 717/102, 717/108, 717/109, 717/113

ABSTRACT:

A system, method and article of manufacture are provided for providing a warning upon retrieval of objects that are incomplete. An object is provided with at least one missing attribute. Upon receipt of a request from an application for the object access to the attributes of the object is allowed by the application. A warning is provided upon an attempt to access the attribute of the object that is missing.

15 Claims, 195 Drawing figures
Exemplary Claim Number: 1
Number of Drawing Sheets: 123

☐ 6. Document ID: US 6442748 B1

L9: Entry 6 of 7 File: USPT Aug 27, 2002

US-PAT-NO: 6442748
DOCUMENT-IDENTIFIER: US 6442748 B1

TITLE: System, method and article of manufacture for a persistent state and persistent object separator in an information services patterns environment

DATE-ISSUED: August 27, 2002

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Bowman-Amuah; Michel K.	Colorado Springs	CO		

US-CL-CURRENT: 717/108; 707/103R, 707/104.1, 707/9, 719/316

ABSTRACT:

A system, method and article of manufacture are provided for separating logic and data access concerns during development of a persistent object for insulating development of business logic from development of data access routine. A persistent object being developed is accessed and a state of the persistent object is detached into a separate state class. The state class serves as a contract between a logic development team and a data access development team. Logic development is limited by the logic development team to developing business logic. Data access development is restricted by the data access development team to providing data creation, retrieval, updating, and deletion capabilities.

18 Claims, 195 Drawing figures
Exemplary Claim Number: 1
Number of Drawing Sheets: 123

☐ 7. Document ID: US 5734907 A

L9: Entry 7 of 7 File: USPT Mar 31, 1998

US-PAT-NO: 5734907
DOCUMENT-IDENTIFIER: US 5734907 A

TITLE: Method of programming an information processing device for network management applications using generic programming

DATE-ISSUED: March 31, 1998

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Jarossay; Mylene	Malakoff			FR
Attal; Denis	Chatillon			FR

US-CL-CURRENT: 717/141; 717/114, 717/139

ABSTRACT:

Method of programming an information processing system having a plurality of data processing devices connected with a network and having access to data stored in complex structures, such as sets or trees, including providing a programming language on said information processing system in which data types of variables used therein are defined by the content of the variables, allowing program variables used with said programming language to be defined without assigning a predetermined data type thereto, developing application programs of the network management type using said programming language, manipulating said complex data structures using said application programs by requesting and receiving data from said complex data structures without designating particular data types of the data to be received, and providing a language interpreter for said programming language on each of said plurality of data processing devices.

8 Claims, 3 Drawing figures
Exemplary Claim Number: 1
Number of Drawing Sheets: 3

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw Desc | Im

Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Terms	Documents
L7 and L4	7

Display Format: - Change Format